Problem D. Fire

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Pang lives on a tree with n vertices. The vertices are labelled as $1, 2, \ldots, n$ and *Pang* is in vertex 1. Each vertex has a temperature. On the morning of each day after day 0, the temperature of each vertex decreases by 1. The temperature doesn't decrease on day 0. On the afternoon of each day, *Pang* can travel to an adjacent vertex, provided that he is at a vertex with positive temperature and his destination vertex has a non-negative temperature. On the evening of each day, if the temperature is higher than or equal to 0, *Pang* can cast magic which increases the temperature of the vertex he is in by k. For each pair of adjacent vertices a and b, *Pang* can travel from vertex a to vertex b at most once (and from b to a at most once). He can choose not to travel and stay in the current vertex.

Pang wants to cast his magic on each vertex exactly once. He also tries to stay at vertex 1 as long as possible, before traveling to any other city. Given the temperature of each vertex right before the morning of the day 1, on which day must *Pang* prepare for departing? If *Pang* prepares on day i, he can cast his magic on that day and will make his first move on day $i + 1$. If he cannot cast his magic on each vertex exactly once even if he prepares for departing on the day 0, output -1.

Input

The first line contains two integers n and k ($2 \le n \le 100000, 0 \le k \le 1000000000$).

Each of the next $n - 1$ lines contains two integers x and y, indicating an edge between vertices x and y ($1 \le x, y \le n$).

The $(n+1)$-th line contains n integers $a_1, a_2, \ldots, a_n$ — the temperature of vertex i right before the morning of day 1 ($0 \le a_i \le 1000000000$).

It's guaranteed that the input is a tree structure.

Output

If he cannot cast his magic on each vertex exactly once, output -1.

Otherwise, output a single integer x — he must prepare for departing from vertex 1 on day x. Day 1 is the day after day 0, and so on.

Examples

standard input	standard output
3 1 1 2 1 3 4 3 5	1
3 1 1 2 1 3 2 10 10	-1